

CLAIMS

1. Control process with management of an opaque user identifier for the complete delivery of a service using at least one server, characterised in that it is done by a transactional identifier server means storing a description of a plurality of
5 service offers taken out by the user from value added service providers, in a memory for each user, the said transactional identifier server means comprising a management module used to associate an opaque transactional identifier with a user or user group and at least one determined service, the process comprising the following steps:

- 10 - an "enabler" server element that intercepted a service request from a user, or one of the said service providers sending an open transaction request of at least one service calling at least one determined "enabler" server element executing sub-transactions, this request being described sequentially with a batch of open primitives sent to a communication interface of the transactional identifier
15 server means and notifying a user identification,
 - analysis of the request and generation of an opaque transactional identifier by management and control means of the transactional identifier server, then,
 - an execution step of the transaction using the opaque transactional
20 identifier.

2. Process according to claim 1, characterised in that the analysis step comprises a check by the management module of the correspondence of determined "enabler" server elements with a listed service offer accessible for the
25 user among the plurality of service offers and a check on the authorization to open the transaction by control means of the transactional identifier server means for the service supplied by the "enabler" server elements and the specified user, particularly as a function of the user identification.

3. Process according to claim 1, characterised in that the execution step of the transaction is initiated by a value added service provider having received the opaque transactional identifier from the transactional identifier server means, the service provider making a request to a determined enabler server element with the opaque transactional identifier as a parameter, for a determined service forming a sub-transaction, to trigger sending an unmask request to the transactional identifier server, in response on the determined enabler server element, to enable the supply of a non-opaque identification number corresponding to the opaque transactional identifier starting from the opaque identifier, followed by a check carried out by the check means of the transactional identifier server means to check if the determined "enabler" server element is or is not authorized for this service and for this user, such that if it is authorized, the non-opaque identification number is transmitted through a communication interface called the enabler interface to the determined server element to enable execution of the sub-transaction.

4. Process according to claim 1, characterised in that the transactional identifier composed of not more than 15 digits, is conform with the UIT-T E-164 numbering plan and the non-opaque identification number is the MSISDN number.

20

5. Process according to claim 1, in which the transactional identifier server means comprises firstly a transactional motor generating emissions of transactional events composed of one of the BEGIN, COMMIT, ROLLBACK commands and secondly a traceability motor recording each event from the transactional motor and all information transmitted during use of the transactional identifier server means in the memory.

25

6. Process according to claim 1, characterised in that the opaque transactional identifier is sent to the service offer supplier after memorisation of a

transactional context in a memory of the transactional identifier server means, indicating particularly:

- a user identification number;
- the transactional identifier;
- 5 - the offer associated with the transaction;
- the state of progress of the transaction for the offer associated with it.

7. Process according to claim 6, in which the opaque transactional identifier is sent to the service offer provider only after a transactional event has been
10 generated representing the start of a transaction to at least one external system, through a second communication interface of the identifier server means called the transaction notification interface.

8. Process according to claim 7, in which a transactional event representing
15 the start of a transaction to at least one external system is generated by a BEGIN command generated by a transactional motor of the transactional identifier server means.

9. Process according to claim 7, in which the transactional identifier server
20 means transmits data representing whether or not the offer is complete from the transaction notification interface to at least one external system, using a COMMIT command generated by the transactional motor to inform the external system for example invoicing system, that the transaction is completely finished.

25 10. Process according to claim 7, in which the transactional identifier server means sends an end of ROLLBACK transactional event through the transaction notification interface to notify at least one external system that the number of transaction rollbacks on error has been exceeded, and that the transaction is cancelled to provide data to a dialog manager and to decide whether or not to
30 invoice this service.

11.Process according to claim 1, in which the management and control means of the transactional identifier server perform the analysis of the open transaction request, particularly by solving correspondence between a technical service address notified in the open transaction request and a listed service offer listed among the various service offer descriptions stored in the memory of the transactional identifier server means.

12.Process according to claim 1, in which the memory of the transactional identifier server means stores service offer descriptions validated by the said suppliers, input through a third communication interface called the service description supply interface.

13.Process according to claim 1, in which the description of a service offer comprises data formulated in a meta language or an equivalent form enabling the control means of the identifier server means to check if the service is being executed correctly and to detect the start and the end.

14.Process according to claim 1, in which the transactional identifier server means comprises an additional communication interface for use by value added service providers, the first interface being for use by server elements.

15.Process according to claim 1, in which the transactional identifier server means comprises internal logic performing the following methods: Start, Completed, Error, Mask, Unmask, Update, Open Transaction, Close Transaction.

16.Process according to claim 15, in which the Start method of the transactional identifier server means generates a transactional identifier, creates a transactional context in memory, generates a BEGIN type transactional event and returns the transactional identifier to the service offer supplier.

17. Process according to claim 15, in which the Completed method of the transactional identifier server means carries out a test to determine if a sub-transaction of the transaction has been executed, modifies the transactional context accordingly, scans the description of the offer to determine if it is necessary
5 for the transactional identifier server means to wait for an external event, sets the logic either waiting for a timeout, or a Close transaction, checks if the transaction is complete and generates a COMMIT type transactional event.

18. Process according to claim 15, in which the Error method of the
10 transactional identifier server means checks if the number of transaction rollbacks on error has been exceeded and if it has, generates a ROLLBACK type transactional event.

19. Process according to claim 15, in which the Mask method is sent by an
15 "enabler" server element to find information for the targeted offer starting from the technical address and the plurality of service offers, to control access of a user subscribing to the service offer and to send either an access refusal or trigger the Start method.

20. Process according to claim 15, in which the Unmask method is sent by
20 an "enabler" server element to find information for the targeted offer starting from data representing the technical address and the transactional identifier, and starting from the said plurality of offers, to control access of a partner provider to the "enabler" server element, to check that the request made to the server element
25 corresponds to the current context of the transaction, and to notify the server element that the transactional identifier server means is waiting for an update, return the MSISDN number associated with the opaque transactional identifier, start waiting for the update, then check that the received update contains the information necessary to execute the offer, to either send a Completed method or
30 an Error method.

21. Process according to claim 15, in which the Update method is sent by an "enabler" server element and consists of putting into the waiting state for an update concerning execution of the request by the transactional identifier server.

5 22. Process according to claim 15, in which the Open Transaction method is sent by a value added service supplier to control access of a partner to one of the operator's subscribers and to generate either an access refusal or to trigger a Start method.

10 23. Process according to claim 17, in which the Close Transaction method is sent by a value added service provider and generates an event capable of unblocking the timeout of the logic of the transactional identifier server means.